MP0064

APPLICATION

FOR

UNITED STATES LETTERS PATENT

Be it known that we, Gregory Burd, citizen of the United States of America and Zining Wu, citizen of the People's Republic of China, have invented new and useful improvements in:

LDPC ENCODER AND METHOD THEREOF

of which the following is the specification.

LDPC ENCODER METHOD THEREOF

Inventors: Gregory Burd

Zining Wu

CROSS REFERENCE TO RELATED APPLICATIONS

The present invention claims priority under 35 U.S.C. §119 (e) from U.S. provisional application serial no. 60/214781, entitled "Address Generator for LDPC Encoder and Decoder and Method Thereof," filed June 28, 2000, the contents of which are incorporated herein by reference.

The present invention is related to the following commonly-assigned, copending applications:

"Multi-Mode Iterative Detector", filed on April 27, 2000 and assigned application Serial No. 09/559186, the contents of which are incorporated herein by reference,

"Address Generator for LDPC Encoder and Decoder and Method Thereof" filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0063), the contents of which are incorporated herein by reference,

"LDPC Decoder and Method Thereof", filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0065), the contents of which are incorporated herein by reference, and

"Parity Check Matrix and Method of Forming Thereof", filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0069), the contents of which are incorporated herein by reference.

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates generally to a linear block encoder in a data transmission system. More particularly, the present invention relates a low-density parity-check code (LDPC) encoder for a write channel in a channel.

Description of the Related Art

Fig. 1 illustrates a conventional digital data transmission system. As shown therein, a digital data transmission system comprises a transmitting section 300 for transmitting user data to receiver 500 via communication channel 401.

The operation of transmission section 300 will now be explained. Prior to processing by transmitting section 300, input or user data maybe encoded with an error correcting code, such as the Reed/Solomon code, or run length limited encoded (RLL) or a combination thereof by encoder 302. The encoded output of encoder 302 is then interleaved by deinterleaver 308 for input to linear block code encoder 304 which generates parity data in a known manner utilizing linear block codes. One example of a linear block code is a low-density parity-check code (LDPC) which is discussed by Robert G. Gallager in *Low-Density Parity-Check Codes*, 1963, M.I.T. Press and by Zining Wu in *Coding and Iterative Detection For Magnetic Recording Channels,* 2000, Kluwer Academic Publishers, the contents of each of which are incorporated in their entirety by reference. Deinterleaver 308 permutes the data so that the same data is reordered before encoding by linear block code encoder 304. By permuting or redistributing the data, interleaver 306 attempts to reduce the number of nearest neighbors of small distance error events. User data at the output of encoder 302 is referred to as being in the channel domain; that is the order in which data is transmitted through the channel. The order of data processed by deinterleaver 308 is referred to as being in the linear block code domain. The parity data from linear block code encoder 304 is combined with the data encoded by encoder 302 by multiplexer 306 for input to channel transmitter 310.

Transmitter 310 transmits the combined user and parity data from multiplexer 306 typically as an analog signal over communication channel 401 in the channel domain. Communication channel 401 may include any wireless, wire, optical and the like communication medium. Receiver 500 comprises an analog to digital converter 502 to convert the data transmitted on communication channel 401 to a digital signal. The digital signal is input to soft channel decoder 504, which provides probability information of the detected data. Soft channel decoder may be implemented by a Soft Viterbi Detector or the like. The output of the soft channel decoder 504, which is in the channel domain, is converted into the linear block code domain by deinterleaver 510. Deinterleaver 510 is constructed similarly to deinterleaver 308. Soft linear block code

decoder 506 utilizes this information and the parity bits to decode the received data. One output of soft linear block code decoder 506 is fed back to soft channel decoder 504 via interleaver 512, which converts data in the linear block code domain to the channel domain. Interleaver 512 is constructed to perform the reverse operations of deinterleaver 510. Soft channel decoder 504 and soft linear block code decoder 506 operate in an iterative manner to decode the detected data.

The other output of soft linear block code decoder 506 is converted from the linear block domain to the channel domain by interleaver 514. Interleaver 514 is constructed similarly to interleaver 512. The output of interleaver 514 is passed on for further processing to decoder 508. Decoder 508 is implemented to perform the reverse operations of encoder 302.

Fig. 2 is a block diagram of a data transmission system implementing an address generator in lieu of the interleave/deinterleaver. A more detailed description of this system can be found in commonly assigned copending application "Address Generator for LDPC Encoder and Decoder and Method Thereof" filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0063), the contents of which are incorporated herein by reference. In general as shown therein, a digital data transmission system comprises a transmitting section 300' for transmitting user data to receiver 500' via communication channel 401. The inventors have observed that a linear block code encoder is not dependent on a position of a bit interleaved. Rather the linear block code encoder only requires a list of equations for a given bit. In other words, there is no need to process the data in the order defined by the interleaver, instead data may be processed in the same order as it is written to the channel. This can be accomplished by incorporating an address generator to provide an address of the appropriate equation of the linear block code encoder. This principle can be similarly applied to the soft linear block decoder. As a result, deinterleaver 308 of the conventional system is now replaced by address generator 328, and deinterleaver 510 is now replaced by address generator 530. Accordingly, there is no requirement for the physical interleaving of data in the receiver 500', since the data remains in the same order as the order of bits of data in the channel throughout this system. The order of bits of data transmitted through the channel is referred to as the channel domain.

A Low-Density Parity-Check Code (LDPC) of block length N (codeword size) has a parity check matrix **H** of size N_p x N and is of full rank (except for two extra rows), and N_p << N. The code space of this code consists of all codewords satisfying:

$$\{c \in \{0,1\}^N \,|\, Hc = 0\} \quad (1),$$

where **c** is a N x 1 vector.

LDPC encoder 304 takes a column of user bits, **u**, having a length of N_u=N-N_p and inserts N_p parity bits to form codeword **c** to satisfy equation 1. A parity vector **p** is combined with **u** in multiplexer 306 to form codeword **c**.

The parity vector **p** is generated by multiplying **u** by a parity generating matrix P having a size of N_u x N_p.

$$\mathbf{p}=P\mathbf{u} \quad (2)$$

For example for N_p=220 and N~5,000, H can be chose to have a nice geometric structure. However, the corresponding parity generating matrix P is very irregular which in hardware would require storing all N user bits in flip-flop registers to perform matrix multiplication by P. and LDPC encoder would require additional area on the integrated circuit. It is estimated that such an LDPC encoder would require approximately 0.66 mm^2.

Summary of the Invention

According to a first aspect of the present invention a method is provided to perform low-density parity-check code encoding of user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that **H•c** =0, comprising the steps of: (a) receiving the user data of block length N_u; (b) decomposing **H•c** into a first component $\mathbf{H_u}$•**u** corresponding to the user data and a second component $\mathbf{H_p}$•**p** corresponding to the parity data such that **H•c** =$\mathbf{H_u}$•**u** + $\mathbf{H_p}$•**p**=0; (c) calculating a vector $\underline{u}$ = $\mathbf{H_u}$•**u**; and (d)calculating p= $\mathbf{H_u}^{-1}$ • $\underline{u}$.

According to a second aspect of the present invention, $\mathbf{H_u}$ comprises a N_p x N_u matrix and $\mathbf{H_p}$ comprises a N_p x N_p matrix.

According to a third aspect of the present invention the method further comprises the step of (e) receiving address information, and step (c) is performed in accordance with step(e).

According to a fourth aspect of the present invention step (c) comprises the step of updating elements of $\underline{u}$ as follows: $\underline{u}(i)=\underline{u}(i) \oplus$ bit.

According to a fifth aspect of the present invention step (d) comprises the step of (g) reducing a row weight of $\mathbf{H_u^{-1}}$ by representing $\mathbf{H_u^{-1}}$ as M1 * M2.

According to a sixth aspect of the present invention step (d) comprises the step of (g) reducing a row weight of $\mathbf{H_u^{-1}}$ by representing $\mathbf{H_u^{-1}}$ as $\prod_{i=1}^{s} M_i$.

According to a seventh aspect of the present invention step (c) is performed prior to step (d).

According to an eighth aspect of the present invention, a low-density parity-check code encoder is provided to encode user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that $\mathbf{H \bullet c} = 0$. An input inputs the user data of block length N_u, an H c decomposer decomposes $\mathbf{H \bullet c}$ into a first component $\mathbf{H_u \bullet u}$ corresponding to the user data and a second component $\mathbf{H_p \bullet p}$ corresponding to the parity data such that $\mathbf{H_u \bullet u} + \mathbf{H_p \bullet p} = 0$. A $\underline{u}$ calculator to calculate a vector $\underline{u} = \mathbf{H_u \bullet u}$, and a p=$\underline{P}$ $\underline{u}$ calculator to calculate $p = \mathbf{H_u^{-1}} \bullet \underline{u}$.

According to a ninth aspect of the present invention, a second input is provided to input address information, and the $\underline{u}$ calculator calculates the vector $\underline{u} = \mathbf{H_u \bullet u}$ in accordance with the second input.

According to a tenth aspect of the present invention, the $\underline{u}$ calculator updates elements of $\underline{u}$ as follows: $\underline{u}(i) = \underline{u}(i) \oplus \text{bit}$.

According to an eleventh aspect of the present invention, the p=$\underline{P}$ $\underline{u}$ calculator reduces a row weight of $\mathbf{H_u^{-1}}$ by representing $\mathbf{H_u^{-1}}$ as M1 * M2.

According to a twelfth aspect of the present invention, the p=$\underline{P}$ $\underline{u}$ calculator reduces a row weight of $\mathbf{H_u^{-1}}$ representing $\mathbf{H_u^{-1}}$ as $\prod_{i=1}^{s} M_i$.

According to a thirteenth aspect of the present invention, the $\underline{u}$ calculator calculates the vector $\underline{u} = \mathbf{H_u \bullet u}$ prior to the p=$\underline{P}$ $\underline{u}$ calculator calculating $p = \mathbf{H_u^{-1}} \bullet \underline{u}$.

According to a fourteenth aspect of the present invention, a computer program is provided to perform low-density parity-check code encoding of user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that $\mathbf{H \bullet c} = 0$, comprising the steps of (a) receiving the user data of block length N_u; (b) decomposing $\mathbf{H \bullet c}$ into a first component $\mathbf{H_u \bullet u}$ corresponding to the user data and a second component $\mathbf{H_p \bullet p}$ corresponding to the

parity data such that $\mathbf{H_u \bullet u} + \mathbf{H_p \bullet p}=0$; (c) calculating a vector $\underline{u} = \mathbf{H_u \bullet u}$; and (d) calculating $p= \mathbf{H_u}^{-1} \bullet \underline{u}$.

According to a fifteenth aspect of the present invention, 22, a data transmission system is provided for transmitting user data to and receiving data from a communication channel. A low-density parity-check code encoder encodes the user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that $\mathbf{H \bullet c} =0$. The encoder comprises an input to input the user data of block length N_u; an H c decomposer to decompose $\mathbf{H \bullet c}$ into a first component $\mathbf{H_u \bullet u}$ corresponding to the user data and a second component $\mathbf{H_p \bullet p}$ corresponding to the parity data such that $\mathbf{H_u \bullet u} + \mathbf{H_p \bullet p}=0$; a $\underline{u}$ calculator to calculate a vector $\underline{u} = \mathbf{H_u \bullet u}$; and a $p=\underline{P}\ \underline{u}$ calculator to calculate $p= \mathbf{H_u}^{-1} \bullet \underline{u}$. A transmitter transmits an output of the low-density parity-check code encoder to the communication channel. A soft channel decoder decodes data from the communication channel, and a soft low-density parity-check code decoder decodes data decoded by the soft channel decoder.

According to a sixteenth aspect of the present invention, a low-density parity-check code encoder encodes user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that $\mathbf{H \bullet c} =0$. An input means is provided for inputting the user data of block length N_u, and an H c decomposer means decomposes $\mathbf{H \bullet c}$ into a first component $\mathbf{H_u \bullet u}$ corresponding to the user data and a second component $\mathbf{H_p \bullet p}$ corresponding to the parity data such that $\mathbf{H_u \bullet u} + \mathbf{H_p \bullet p}=0$. A $\underline{u}$ calculating means for calculating a vector $\underline{u} = \mathbf{H_u \bullet u}$, and a $p=\underline{P}\ \underline{u}$ calculating means for calculating $p= \mathbf{H_u}^{-1} \bullet \underline{u}$.

According to a seventeenth aspect of the present invention a second input means is provided for inputting address information, and the $\underline{u}$ calculating means calculates the vector $\underline{u} = \mathbf{H_u \bullet u}$ in accordance with the second input means.

According to an eighteenth aspect of the present invention, the $\underline{u}$ calculating means updates elements of $\underline{u}$ as follows: $\underline{u}(i)=\underline{u}(i) \oplus \text{bit}$.

According to a nineteenth aspect of the present invention, the $p=\underline{P}\ \underline{u}$ calculating means reduces a row weight of $\mathbf{H_u}^{-1}$ by representing $\mathbf{H_u}^{-1}$ as M1 * M2.

According to a twentieth aspect of the present invention, the $p=\underline{P}\ \underline{u}$ calculating means reduces a row weight of $\mathbf{H_u}^{-1}$ representing $\mathbf{H_u}^{-1}$ as $\prod_{i=1}^{s} M_i$.

According to a twenty-first aspect of the present invention, the u calculating means calculates the vector u = $\mathbf{H_u \bullet u}$ prior to the p=P u calculating means calculating p= $\mathbf{H_u}^{-1} \bullet$ u.

According to a twenty-second aspect of the present invention, a data transmission system is provide for transmitting user data to and receiving data from a communication channel. A low-density parity-check code encoding means encodes user data u of length N_u, by inserting parity data p of length N_p into output data c of length N in accordance with a parity matrix H such that $\mathbf{H \bullet c}$ =0, comprising, and an input means inputs the user data of block length N_u. A H c decomposer means is provided for decomposing $\mathbf{H \bullet c}$ into a first component $\mathbf{H_u \bullet u}$ corresponding to the user data and a second component $\mathbf{H_p \bullet p}$ corresponding to the parity data such that $\mathbf{H_u \bullet u} + \mathbf{H_p \bullet p}$=0. A u calculating means calculates a vector u = $\mathbf{H_u \bullet u}$, and a p=P u calculating means for calculates p= $\mathbf{H_u}^{-1} \bullet$ u. A transmitting means transmits an output of the low-density parity-check code encoding means to the communication channel. A soft channel decoding means decodes data from the communication channel, and a soft low-density parity-check code decoding means decodes data decoded by the soft channel decoding means.

Other objects and attainments together with a fuller understanding of the invention will become apparent and appreciated by referring to the following description and claims taken in conjunction with the accompanying drawings.

Brief Description of the Drawings

In the drawings wherein like reference symbols refer to like parts.

Fig. 1 is a block diagram of a data transmission system;

Fig. 2 is a block diagram of another data transmission;

Fig. 3 is a block diagram of a data transmission system in accordance with the present invention;

Fig. 4 is an example of a parity check matrix in accordance with the present invention; and

Fig. 5 is a block diagram of a low-density parity-check code encoder in accordance with the present invention.

Description of the Preferred Embodiments

Reference is now made to Fig. 3, which is a block diagram of a data transmission system in accordance with the present invention. In general as shown therein, a digital data transmission system comprises a transmitting section 300' for transmitting user data to receiver 500' via communication channel 401. The operation of transmission section 300' will now be explained. Prior to processing by transmitting section 300', as in the conventional system, input or user data maybe encoded with an error correcting code, such as the Reed/Solomon code, or run length limited encoded (RLL) or a combination thereof by encoder 302. User data **u** is temporarily stored in memory 382, preferably implemented as SRAM. Addresses for the parity equations of linear block code encoder 304 are generated by address generator. Address generator 328 is responsive to counter 730 under the control of controller 740. Controller 740 synchronizes counter 730 to the output of encoder 302 so that counter 730 can provide a count of the number of bits in a data block output by encoder 302. In the preferred embodiment the data block size is approximately 5000 bits.

Linear block code encoder 304 utilizes the user data and addresses from address generator 328 to provide the parity bits to multiplexer 306. Linear block code encoder 304 is preferably implemented as a low-density parity-check code (LDPC). The parity data from linear block code encoder 304 is combined with the user data **u** stored in SRAM 382.

Transmitter 310 transmits the combined user and parity data from multiplexer 306 typically as an analog signal over communication channel 401 in the channel domain. Communication channel 401 may include any wireless, wire, optical, magnetic channel and the like.

Receiver 500' comprises an analog to digital converter 502 to convert the data transmitted on communication channel 401 to a digital signal. The digital signal is input to soft channel decoder 504, which provides soft or probabilistic information of the detected data to soft linear block decoder 506. Soft channel decoder may be implemented as a Soft Viterbi Detector or the like, and address generator 530 may be constructed similarly as address generator 328 in transmission section 300'. The soft information output by soft channel decoder 504 remains in the channel domain and is decoded by soft linear block code decoder 506, in accordance with the address of the parity equations generated by address generator 530. Address generator 530 is responsive to counter 735 under the control of controller 745. Controller 745

synchronizes counter 735 to the output of soft channel decoder 504 so that counter 735 can provide a count of the number of bits in a codeword output by soft channel decoder 504 and a count of the number of codewords.

Soft linear block code decoder 506 operates in combination with soft channel decoder 504 and address generator 530 in an iterative fashion. Soft linear block code decoder is preferably implemented as a low-density parity- check code (LDPC) decoder as described in commonly assigned, copending patent application entitled "LDPC Decoder and Method Thereof", filed on even date and assigned application Serial No. _________ (Attorney Docket No. MP0065), the contents of which are incorporated herein by reference.

After the iterative process has completed, the output of soft linear block code decoder 506 is passed on for further processing to decoder 508. Decoder 508 is implemented to perform the reverse operations of encoder 302 or correct for any data errors.

As noted above, the parity data **p** is inserted into user data **u** by means of multiplexer 306 to form the codeword **c** as an N x 1 vector. This can be connoted as **c**≡ [**u**,**p**]. As will be appreciated by one of ordinary skill in the art, the columns of H can be simply rearranged so that the last N_p columns of H correspond to the parity bits. The calculation of the parity bits is equivalent to the solving of a system of linear equations

$$\mathbf{H}\bullet[\mathbf{u},\mathbf{p}]=0 \quad (3)$$

for the parity vector **p**. The linear system in equation (3) can be rewritten as:

$$\mathbf{H_u}\bullet\mathbf{u} + \mathbf{H_p}\bullet\mathbf{p}=0 \quad (4),$$

where H_u is an N_p x N_u matrix consisting of the first N_u columns of H and H_p is a N_p x N_p matrix consisting of the parity columns. The H c decomposer 580 performs the rewriting or decomposition of **H•c**. It is noted that the first term of equation (4), namely $\mathbf{H_u}\bullet\mathbf{u}$, depends only on the user data and can be precomputed utilizing address generator 328 as explained in detail hereinbelow for storage in preferably flip-flop registers.

Let $\underline{u}$ be a N_p x 1 vector, where $\underline{u} = \mathbf{H_u}\bullet\mathbf{u}$, and substituting this expression into equation (4) becomes:

$$\mathbf{H_p}\bullet\mathbf{p}=\underline{u} \quad (5)$$

(Since u̲ is a binary vector, u̲=-u̲)

Since matrix H_p is of full rank, it is invertible. P̲ is defined as H_p^{-1}, and the solution to equation (3) is:

$$\mathbf{p}=\underline{P}\bullet\underline{u} \qquad (6)$$

In view of the above derivation, the encoding procedure can be separated into two steps. First u̲ is calculated by u̲ calculator 582 from the user bits **u** utilizing address information from address generator 328, and second the shortened encoding matrix P̲ is used to obtain the parity vector **p**. It is noted that the first step is relatively easy to calculate and the second step still requires a matrix multiplication. However P̲ is a N_p x N_p matrix which is relatively sparse, whereas in the conventional arrangement P is a N_p x N_u matrix. As will be shown herein below, P̲ has an average row weight of approximately 24. This is in contrast to a row weight of approximately 105 for matrix P. It is noted that the complexity of matrix multiplication is determined by a matrix's sparseness, rather than a matrix's dimension. Thus in accordance with the present invention, approximately (105-24) N_p exclusive OR (XOR) operations are saved for each LDPC parity bit.

For example if H is a 3 x 9 matrix as follows and the underlying interleaver is I(i)=i

u1	u2	u3	u4	u5	u6	p1	p2	p3
1	1	1	0	0	0	1	0	0
0	0	0	1	1	1	0	1	0
0	0	1	1	0	0	0	0	1

(7)

H_u comprises the first 6 columns of H as follows:

b1	b2	b3	b4	b5	b6
1	1	1	0	0	0
0	0	0	1	1	1
0	0	1	1	0	0

(8)

H_p comprises the last 3 columns of H as follows:

p1	p2	p3
1	0	0
0	1	0
0	0	1

(9)

Suppose the input to encoder 304 is:

1
1
0
0
1
1

(10)

If H_u is multiplied by **u**, then u=

0
0
0

(11)

Since H_p is an identity matrix, P is also an identity matrix. Therefor **p**=Pu= u=

0
0
0

(12)

The process of calculating u is as follows. As data is provided from encoder 302 to linear block code encoder 304, the process begins to calculate the parity data. In other words, it is not necessary to have the entire codeword to begin calculating the

parity data. As the data is provided from encoder 302 to linear block code encoder 304, address generator 328 provides row information indicating the equation utilized by the user data. Commonly assigned, copending application entitled, "Address Generator for LDPC Encoder and Decoder and Method Thereof" filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0063), the contents of which are incorporated herein by reference, provides a detailed description of address generator 328.

For each codeword, vector u is initialized to a zero vector, and the components of u are updated as follows:

$$\underline{u}(i)=\underline{u}(i) \oplus \text{bit} \quad (13)$$

As will be apparent to one of ordinary skill in the art, if a user bit is 0, then no processing needs to be performed. Additionally , address generator 328 is adjusted for the skipped parity positions.

The following is an example in accordance with the present invention. First for each codeword, vector u is initialized a zero vector. As user bits are provided to encoder 304, address generator provides the row information. In this example, matrix H_u is set forth in equation (14) below:

	u6	u1	u4	u3	u5	u2
Equation 1	1	1	1	0	0	0
Equation 2	0	0	0	1	1	1
Equation 3	0	0	1	1	0	0

(14)

where the user data is input in order of u1, u2, u3, u4, u5 and u6.

H_p is as follows:

p1	p2	p3
1	0	0
0	1	0
0	0	1

(15)

H_u and H_p of this example satisfy equations (3) and (4).

The equations can be rewritten as:

$$u_6 + u_1 + u_4 + p_1 = 0,$$

$$u_3 + u_5 + u_2 + p_2 = 0, \text{ and}$$

$$u_4 + u_3 + p_3 = 0,$$

(16)

where $\underline{u}_1 = u_6 + u_1 + u_4$; $\underline{u}_2 = u_3 + u_5 + u_2$; and $\underline{u}_3 = u_4 + u_3$

First, initialize $\underline{u}$ =

$$\begin{array}{|c|}\hline 0 \\ \hline 0 \\ \hline 0 \\ \hline \end{array}$$

(17)

Next input u_1. For u_1, address generator 328 outputs r=1. In other words equation 1 of matrix H_u checks the parity of the u_1. In this example, u_1 is assigned 1 and

$\underline{u}$ =

$$\begin{array}{|c|}\hline 0 + 1 \\ \hline 0 \\ \hline 0 \\ \hline \end{array}$$

or

$\underline{u}$ =

$$\begin{array}{|c|}\hline 1 \\ \hline 0 \\ \hline 0 \\ \hline \end{array}$$

(18)

As the next bit u_2 is input, address processor 328 sets r =2, and for u_2=1

$\underline{u}$ =

$$\begin{array}{|c|}\hline 1 \\ \hline 0+1 \\ \hline 0 \\ \hline\end{array}$$

or

$\underline{u}$ =

$$\begin{array}{|c|}\hline 1 \\ \hline 1 \\ \hline 0 \\ \hline\end{array}$$

(19)

The next bit u_3 is input and for u_3=0, no update of $\underline{u}$ occurs. (It will be apparent to one of ordinary skill in the art if u_3, for u_3=0, were to be processed, then $\underline{u}$ would still not change from the previous step, thus resulting in unnecessary processing.)

$\underline{u}$ =

$$\begin{array}{|c|}\hline 1 \\ \hline 1 \\ \hline 0 \\ \hline\end{array}$$

(20)

Similarly for the next bit u_4, for u_4=0, no update of $\underline{u}$ occurs.

$\underline{u}$ =

$$\begin{array}{|c|}\hline 1 \\ \hline 1 \\ \hline 0 \\ \hline\end{array}$$

(21)

As the next bit u_5 is processed, address processor 328 sets r =2, and for u_5=1.

$\underline{u}$ =

1
1+1
0

or

$\underline{u}$ =

1
0
0

(22)

As the next bit u_6 is processed, address processor 328 sets r =1, and for u_6=1.

$\underline{u}$ =

1+1
0
0

or

$\underline{u}$ =

0
0
0

(23)

Once the vector $\underline{u}$ has been computed, some rows are removed because H is not of full rank. In the preferred embodiment, as shown in Fig. 4, the parity check matrix comprises 222 rows (or equations) by 5402 columns, which comprises 220 linearly independent rows (where 5402 =73*74). The matrix can be divided into three tiers of equations having 73, 74 and 75 equations, respectively. As can be seen the tiers (73, 74 and 75) are mutually prime. The set of independent rows can be obtained by removing the last row of the second tier and third tier, namely the 147th row and the 222nd row. The following table shows the values of the elements in the matrix:

Tier	i^{th} position	i^{th} position
1	1 if r = i(mod73)	0 if r ≠ i(mod73)
2	1 if r = i(mod74)	0 if r≠ i(mod74)
3	1 if r = i(mod75)	0 if r ≠ i(mod75)

where r is the index within a tier.

A matrix having 5402 columns can process a maximum LDPC codeword of 5402 bits. A further discussion on the details of the parity check matrix is provided in "Parity Check Matrix and Method of Forming Thereof", filed on even date and assigned application Serial No. __________ (Attorney Docket No. MP0069), the contents of which are incorporated herein by reference.

Utilizing equation (6) the parity vector can be calculated by p=$\underline{P}$ $\underline{u}$ calculator 584 as follows. As noted above a matrix's sparseness, rather than a matrix's dimension determine the complexity of matrix multiplication. One way to quantify the complexity of the matrix is to determine the average row weight of $\underline{P}$. $\underline{P}$ can be decomposed into two (2) matrices M1 and M2, such that $\underline{P}$ = M1 * M2. The preferred method of decomposing $\underline{P}$ is by placing the system into echelon form. In the preferred embodiment, M1 and M2 are each 220 x 220 matrices. The combined row weight of M1 and M2, which is ~24, is lower than that of $\underline{P}$. In general $\underline{P}$ can be represented as $\prod_{i=1}^{s} M_i$ in order to reduce the combined row weight. In accordance with the present invention the area of the encoder is about 0.4 mm² (as compared to 0.66 mm² for conventional encoders).

While the invention has been described in conjunction with several specific embodiments, it is evident to those skilled in the art that many further alternatives, modifications and variations will be apparent in light of the foregoing description. More specifically, while the present invention is preferably implemented as an integrated circuit, it is contemplated that the present invention may also be implemented as discrete components or a general-purpose processor operated in accordance with program code instructions or computer program or combination thereof. These program code instructions can be obtain from a medium, such as network, local area network, the Internet, or storage devices. Such storage devices include, by way of example, magnetic storage devices, optical storage devices, electronic storage devices, magneto-optical device and the like. Thus, the invention

described herein is intended to embrace all such alternatives, modifications, applications and variations as may fall within the spirit and scope of the appended claims.